3D Turbulent Premixed Combustion:
An adaptive low Mach number approach

Marc Day

MSDay@lbl.gov
Center for Computational Sciences and Engineering
Lawrence Berkeley National Laboratory, USA
http://seesar.lbl.gov/ccse/

Presented at: COMREF 2002, 4 December 2002
Computational Methods for Multidimensional Reactive Flows
Heidelberg, Germany

Collaborators

CCSE - Internal

- A. Almgren
- V. Beckner
- J. Bell
- J. Grcar
- M. Lijewski
- C. Rendleman

External

- R. Cheng, I. Shepherd (LBNL) - Premixed flame experiments
- C. Schulz, W. Bessler (U. Heidelberg) - Flame diagnostics
- S. Woosley, M. Zingale (UCSC) - Type 1a supernovae
- P. Glarborg, A. Jensen (DTU Denmark) - Combustion chemistry
- R. Kee, N. Sullivan (CSM) - Experiments/chemistry
- M. Minion (UNC) - Chemistry/CFD splitting methods
- C. Rutland (UWM) - Turbulent reacting sprays
- N. Brown, S. Tonse (LBNL) - Combustion Chemistry
- A. Lutz (SNL) - Chemistry
- D. Goodwin (CalTech) - Chemistry interface

Laboratory-scale low-speed combustion

Objective: Without subgrid models, compute effects of turbulence on chemistry

Application: Pollutant (NO_x) formation in turbulent laboratory flames

Premixed Low-Swirl Burner
(courtesy R. Cheng, LBNL)

Relevant scales:

- Domain size: $\mathcal{O}(10$ cm$)$
- Time scale: $\mathcal{O}(0.1 - 1.0$ s$)$
- Flame thickness: $\mathcal{O}(0.1$ cm$)$
- Sound speed: $\mathcal{O}(10^5$ cm/s$)$

Turbulence/Chemistry Simulations

Options:

1. Turbulence/chemistry subgrid models
   - By definition, interaction details already inside the model
   - Model validation is the objective of the work

2. Turbulent DNS
   (a) Compressible
   - CFL + flame resolution $\Rightarrow \mathcal{O}(10^9)$ zones $\times$ $\mathcal{O}(10^6)$ timesteps
     Appears to require extraordinarily large computing hardware

   (b) Low Mach formulation
   - Fully-implicit $\Rightarrow$ very large matrix sizes
   - <u>Sequential, semi-implicit</u> $\Rightarrow$ complex algorithms, but feasible

Outline of Talk

An adaptive low Mach number algorithm

- Low Mach number model: evolving a constrained velocity field
- The base algorithm (aka: “the single-grid method”)
- Extensions for AMR (Adaptive mesh refinement)
  - Hierarchical block-structured grids
  - Temporal subcycling
  - Synchronizations
- Validation, performance
- Active research

Low Mach Number Combustion

Low Mach number model, $M = U/c \ll 1$ (Rehm & Baum 1978, Majda & Sethian 1985)

$$p(\vec{x},t) = p_0(t) + \pi(\vec{x},t) \quad \text{where} \quad \pi/p_0 \sim \mathcal{O}(M^2)$$

- p_0 does not affect local dynamics, π does not affect thermodynamics
- Acoustic waves analytically removed (or, have been “relaxed” away)
- $\vec{U}$ satisfies a divergence constraint, $\nabla \cdot \vec{U} = S$

Conservation equations:

$$\frac{\partial \rho Y_\ell}{\partial t} + \nabla \cdot \left(\rho Y_\ell \vec{U}\right) = \nabla \cdot \vec{F}_\ell + \rho \dot{\omega}_\ell$$

$$\rho \frac{D\vec{U}}{Dt} + \nabla \pi = \nabla \cdot \tau$$

$$\frac{\partial \rho h}{\partial t} + \nabla \cdot \left(\rho h \vec{U}\right) = \nabla \cdot \vec{Q}$$

- Y_ℓ mass fraction
  $$\sum Y_\ell = 1,\ 0 \le Y_\ell \le 1$$
- $\vec{F}_\ell$ species diffusion, $\sum \vec{F}_\ell = 0$
- $\dot{\omega}_\ell$ species production, $\sum \dot{\omega}_\ell = 0$
- h enthalpy $h = \sum Y_\ell h_\ell(T)$
- $\vec{Q}$ heat flux

Projection Methods

Given an equation-of-state, $p = p(\rho, Y_\ell, T)$ the conservation equations determine the (global) constraint on $\vec{U}$. For an ideal gas

$$\nabla \cdot \vec{U} = S = \frac{1}{T}\frac{DT}{Dt} + \bar{W} \sum \frac{1}{W_\ell}\frac{DY_\ell}{Dt}$$

Fractional-step scheme for constrained flows

1. Advance velocity ignoring constraint using a lagged pressure gradient: $\vec{U}^{n+1,*}$
2. Decompose $\vec{U}^{n+1,*}$ orthogonally to extract the component satisfying the divergence constraint. The remainder is used to update the pressure

Vector field decomposition, any vector $\vec{V}$

$$\vec{V} = \vec{V}_d + \frac{1}{\rho}\nabla\phi \quad \text{(where } \nabla \cdot \vec{V}_d = S\text{)}$$

The scalar, ϕ, satisfies the elliptic equation

$$\nabla \cdot \left(\frac{1}{\rho}\nabla\phi\right) = \nabla \cdot \vec{V} - S$$

This variable-coefficient linear system must be inverted at each time step

Algorithm Overview

Second-order (time & space) integration scheme

Algorithm Components

Data:	Cell-centered, uniform grid
Advection:	Explicit Godunov
Diffusion:	Crank-Nicolson
Projection:	ρ-weighted projection for elliptic constraint
Chemistry:	Stiff ODE integrator (VODE)

Specialization to low Mach flow

- Advective flux incorporates divergence constraint
- Species diffusive fluxes conserve mass
- Chemical reaction terms incorporated using Strang-splitting to avoid globally coupled stiff nonlinear solve

Godunov Advection Details

Conservative scheme requires fluxes computed on cell faces

- Second-order advection flux $\boxed{\Gamma_{\mathsf{A}}^{n+1/2} = U^{n+1/2}\ \phi^{n+1/2}}$
- Slope-limited extrapolatation of $\phi,\ \vec{U}$ from centers on either side

$$\phi_{\mathsf{FACE}}^{n+1/2} = \phi_{\mathsf{CC}}^{n} \pm \frac{\Delta x}{2}\left(\frac{\partial \phi}{\partial x}\right) + \frac{\Delta t}{2}\left(\frac{\partial \phi}{\partial t}\right)$$

- Use PDE to write time-derivative in terms of space-derivative at t^n
- Project time-centered velocity on faces to satisfy $\nabla \cdot \vec{U}^{n+1/2} = S^{n+1/2}$
- "Riemann problem" for $\phi^{n+1/2}$ (no acoustics here, a simple upwind)

NOTE: Characteristics-based methods, such as Godunov, perform best near CFL≈1; determines typical Δt for our simulations

Crank-Nicolson Details

Conservative (edge-based) diffusion flux $\boxed{\Gamma_{\mathsf{D}} = -D\nabla\phi}$

- Semi-implicit diffusion, trapezoidal in time, $\Gamma_{\mathsf{D}} = \left(\Gamma_{\mathsf{D}}^{n} + \Gamma_{\mathsf{D}}^{n+1}\right)/2$
- Transport coefficients, μ, κ, D_ℓ from TRANLIB or EGLib
- Conservation $\Rightarrow \sum \Gamma_{D,\ell} \equiv 0$ via “correction velocity”
- At CFL≈1, well-conditioned linear solves, “a few” multigrid iterations
- A simple 2-pass (predictor/corrector) iteration accommodates variable μ, κ, D_ℓ to second-order in time
- Implicit treatment minimizes expensive property evaluation

Chemistry Details

ODE system

$$\frac{\partial Y_\ell}{\partial t} = \dot{\omega}_\ell \qquad \Longrightarrow \qquad \frac{\partial\,[X_\ell]}{\partial t} = \sum (\nu_{\ell j}^B - \nu_{\ell j}^F) q_j$$

Evolved subject to constant ρ and h. Rate of forward progress of reaction j

$$q_j \quad = \quad k_j^F \prod [X_m]^{\nu_{mj}^F} - k_j^B \prod [X_m]^{\nu_{mj}^R}$$

$$\text{where} \qquad k_j^F = A_j T^{\beta_j} \exp\left(-\frac{E_j}{R_c T}\right), \quad k_j^R = k_j^F / K_{cj}$$

$[X_m]$ is the molar concentration of species m, K_{cj} is the equilibrium constant for reaction j.

$\nu_{mj}^F, \nu_{mj}^R, K_{cj}, A_j, \beta_j, E_j$ *specified via ChemKin database*

Algorithm Summary

Beginnning with the state at t^n

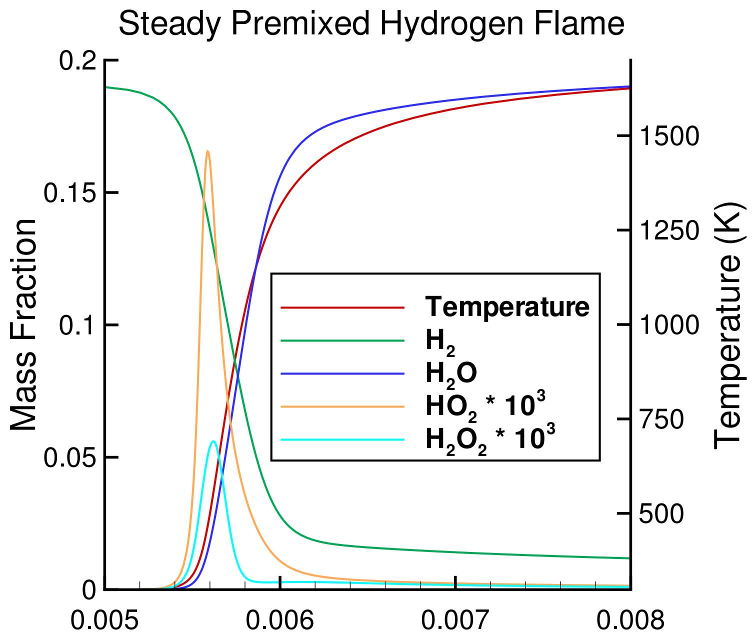
1.	$\vec{U}^{n+1/2}$	Predict, project advection velocity solve variable-ρ elliptic problem (multigrid)
2.	*chemistry* $\Delta t/2$	Stiff ODEs, constant (ρ, h)*
3.	$\Gamma_{\mathsf{A}}^{n+1/2}$	Advection fluxes, time-explicit based on post-chemistry state
4.	$\Gamma_{\mathsf{D}}^{n},\ \Gamma_{\mathsf{D}}^{n+1}$	Diffusion fluxes, implicit parabolic solve (multigrid), predictor/corrector
5.	*chemistry* $\Delta t/2$	Stiff ODEs, constant (ρ, h)*
6.	$(\vec{U}, \pi)^{n+1}$	Predict, project cell-centered velocity solve variable-ρ elliptic solve (multigrid)

* Strang-split chemistry

Convergence

- Freely propagating 1D laminar hydrogen flame
- GRI-Mech 1.2 chemistry, transport, thermodynamics (9 species, 27 reactions)
- Mixture model for diffusion, no radiation, Dufour, Soret or body forces

Quantity	Convergence Rate		
	L^1	L^2	L^∞
T	2.2	2.2	1.7
V	3.9	3.9	3.3
H	2.3	2.2	2.2
ρ	2.1	2.1	2.2
Y_{H2}	2.1	2.0	1.8
Y_H	3.0	2.9	2.5
Y_O	2.6	2.5	2.4
Y_{O2}	1.9	2.1	2.0
Y_{OH}	3.0	3.0	2.3
Y_{H2O}	1.9	2.0	1.8
Y_{HO2}	1.4	1.1	0.7
Y_{H2O2}	1.9	1.9	1.5
Y_{N2}	1.9	2.1	2.3



Essentially second-order

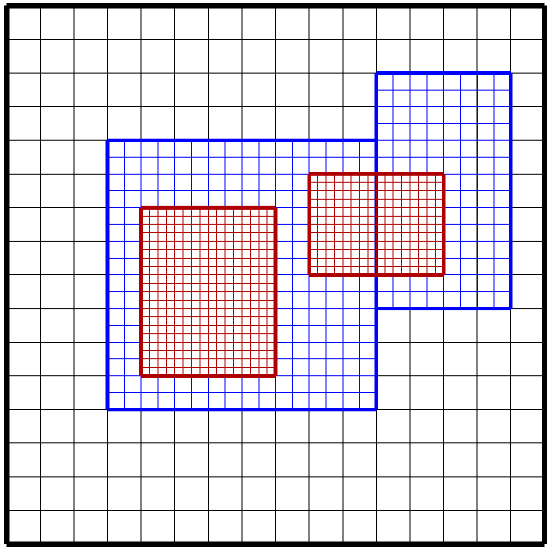
Adaptive Mesh Refinement (AMR)

Observation	In many flame scenarios, resolution requirements are determined by a flame structure that fills only a small fraction of the domain necessary to properly capture related fluid dynamical effects
Goal	Simulate both scales (flame/fluid) with sufficient fidelity to explore their interaction
Approach	Nested hierarchy of block-structured uniform grids

Discussion

1. Grid structure - spatial discretization, refinement levels
2. Grid advance - temporal discretization, subcycling
3. Synchronization - local/global matching conditions

AMR - Grid Structure



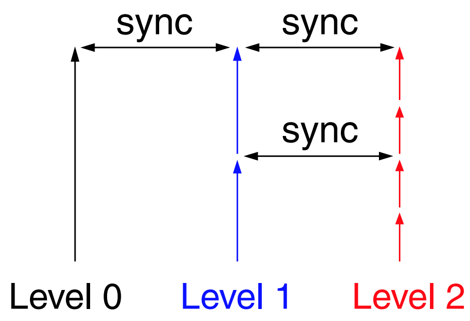
2D adaptive grid hierarchy

Block-structured hierarchical grids

Each grid patch (2D or 3D)

- Logically structured, rectangular
- Refined in space and time by evenly dividing coarse grid cells
- Dynamically created/destroyed to track time-dependent features

Subcycling:

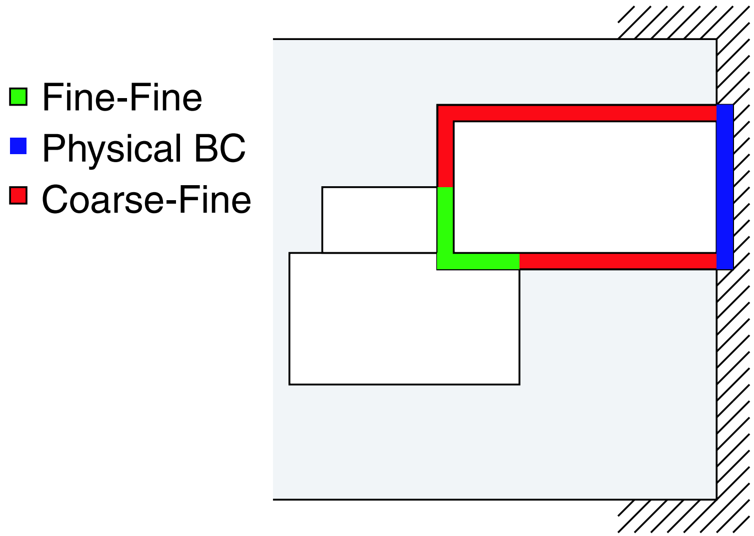


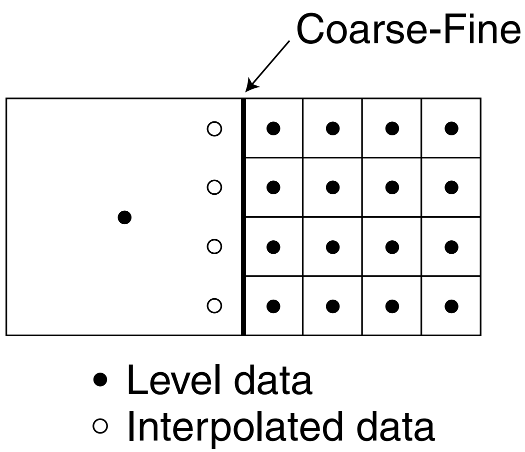
- Advance level ℓ, then
  - Advance level $\ell + 1$
    level ℓ supplies boundary data
  - Synchronize levels ℓ and $\ell + 1$

Preserves properties of advection algorithm while minimizing coarse-grid integration costs

AMR level operations

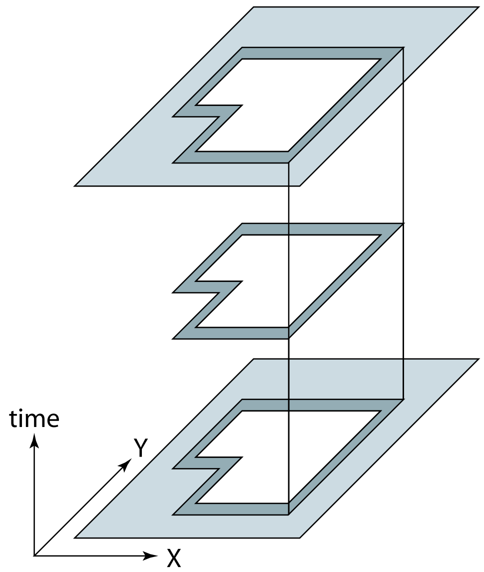
Organize grids by refinement level, couple through “ghost” cells





On the coarse-fine interface:

- Fine: Boundary cells filled from coarse data
  - Interpolated in space and time
- Coarse: Incorporate improved fine solution
  - “Synchronization” (discussed next)



Synchronization: Hyperbolic PDEs

The hyperbolic synchronization is simplest to illustrate

What does it take to build a conservative integration on our grid hierarchy?

Goal of sync: Fix mismatch associated with advancing solution on coarse and fine levels independently

1. Coarse cells covered by fine levels don't have the most accurate data
2. Coarse and fine solution computed with different fluxes

A synchronization that fixes these problems:

- Average fine data onto coarse grids below
- "Reflux" to enforce conservation along the coarse-fine interface

$$\phi^{n+1} \leftarrow \phi^{n+1} - \frac{\Delta t_C}{\Delta x_C}\Gamma_A^C + \sum_t \sum_{\delta\Omega} \frac{\Delta t_F}{\Delta x_F}\Gamma_A^F$$

A simple (explicit) update to coarse solution in cells bounding the fine grids

AMR for Elliptic Equations

How to solve an elliptic equation on a hierarchical mesh system?

$$\mathcal{L}^{c-f}\phi^{c-f} = f^{c-f}$$

Subcycling gives us pieces, how do we put them together?

1. $\mathcal{L}^c\bar{\phi}^c = f^c$ on coarse grids
2. $\mathcal{L}^f\bar{\phi}^f = f^f$ on fine grids, with Dirichlet values from coarse

Solve for the "composite" increment, $\delta\phi$

$$\mathcal{L}(\delta\phi) = f^{c-f} - \mathcal{L}^{c-f}\bar{\phi}^{c-f} = g$$

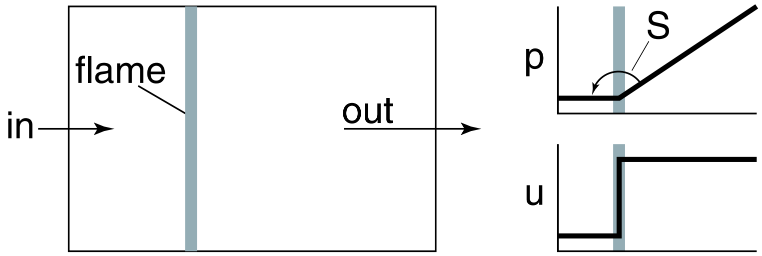
The residual g is localized on the coarse-fine boundary, and $\delta\phi$ is exactly the correction to $\bar{\phi}^c$ and $\bar{\phi}^f$ required to satisfy the composite equation.

For time-dependent applications, the integral of g over time represents a residual due to subcycling.

Composite Projection - An Example

Expansion in 1D flame generates pressure gradient, accelerates flow

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = S\,\delta(x), \quad \rho u \sim \nabla p$$



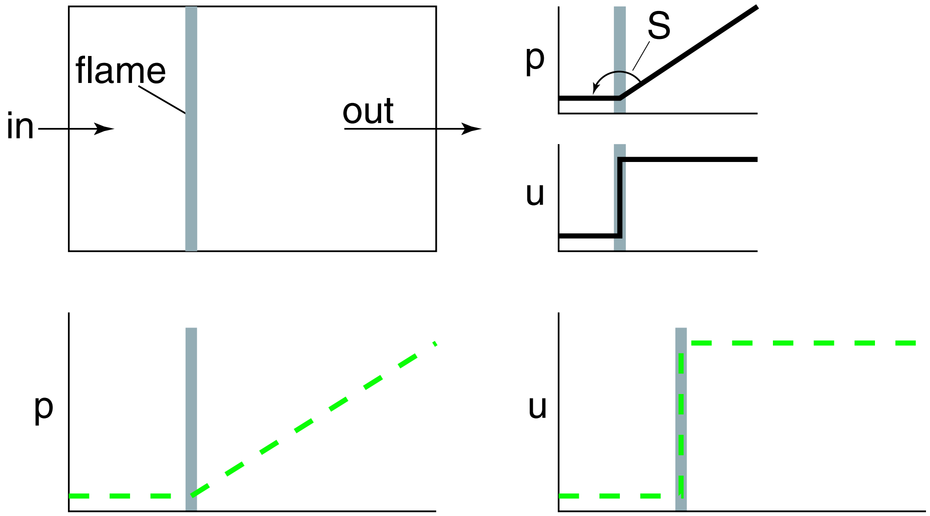
Compute 2 solutions: c,f

- $\Delta x_c > \Delta x_f$
- $S_c < S_f \equiv S$ (ie. S_f just captures S)

Composite Projection - An Example

Expansion in 1D flame generates pressure gradient, accelerates flow

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = S\,\delta(x), \quad \rho u \sim \nabla p$$



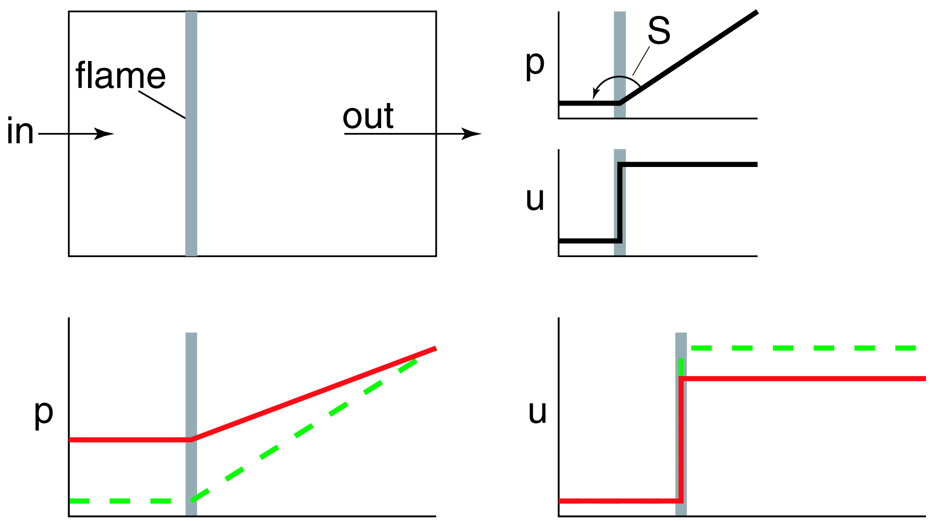
Compute 2 solutions: c,f

- $\Delta x_c > \Delta x_f$
- $S_c < S_f \equiv S$
  (ie. S_f just captures S)
- $p_f,\ u_f$ exact

Composite Projection - An Example

Expansion in 1D flame generates pressure gradient, accelerates flow

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = S\,\delta(x), \quad \rho u \sim \nabla p$$



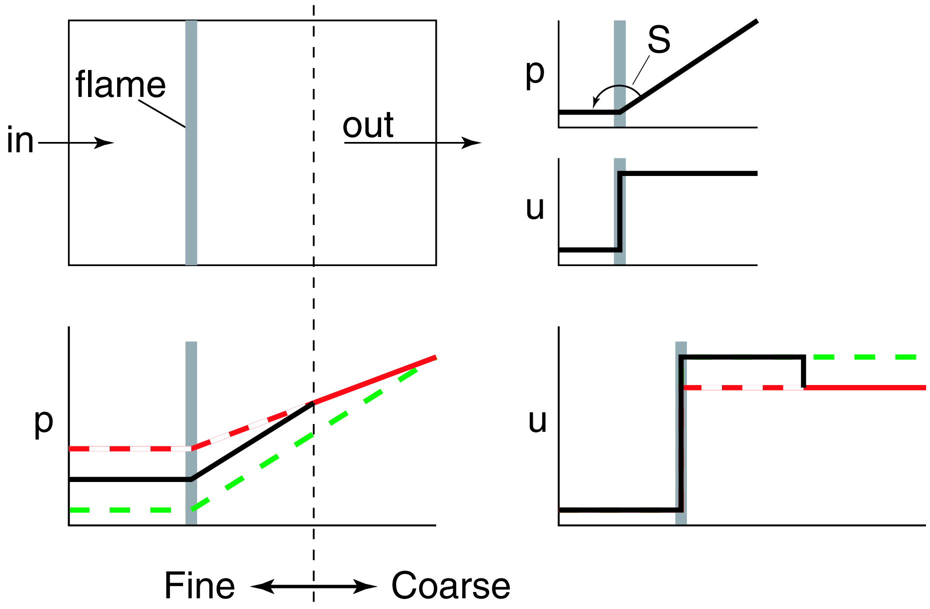
Compute 2 solutions: c,f

- $\Delta x_c > \Delta x_f$
- $S_c < S_f \equiv S$
  (ie. S_f just captures S)
- $p_f,\ u_f$ exact
- Outflow u_c too small

Composite Projection - An Example

Expansion in 1D flame generates pressure gradient, accelerates flow

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = S\,\delta(x), \quad \rho u \sim \nabla p$$



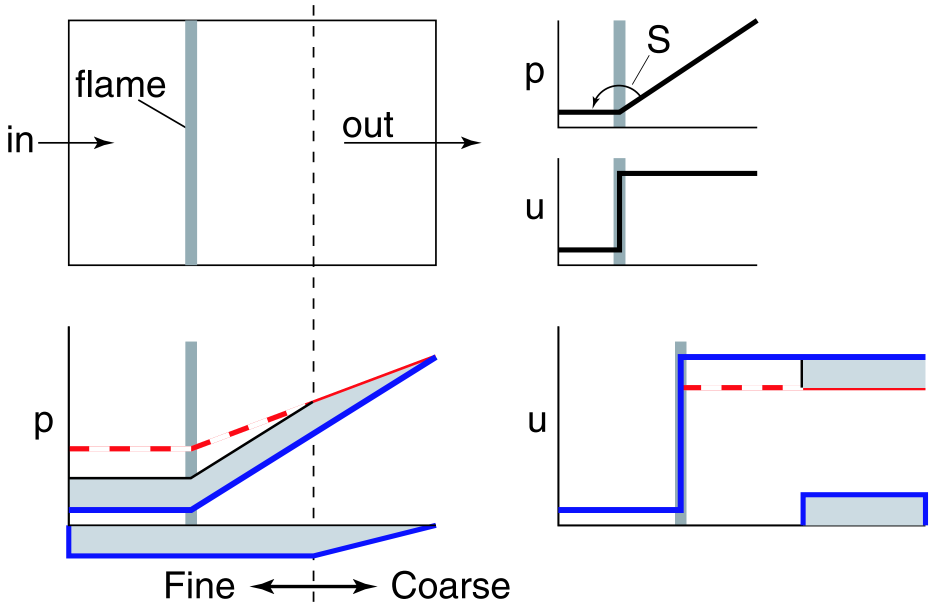
Compute 2 solutions: c,f

- $\Delta x_c > \Delta x_f$
- $S_c < S_f \equiv S$
  (ie. S_f just captures S)
- $p_f,\ u_f$ exact
- Outflow u_c too small
- Jump in u_f ok, but...

Composite Projection - An Example

Expansion in 1D flame generates pressure gradient, accelerates flow

$$\nabla \cdot \left(\frac{1}{\rho} \nabla p \right) = S\,\delta(x), \quad \rho u \sim \nabla p$$



Compute 2 solutions: c,f

- $\Delta x_c > \Delta x_f$
- $S_c < S_f \equiv S$
  (ie. S_f just captures S)
- $p_f,\ u_f$ exact
- Outflow u_c too small
- Jump in u_f ok, but...
- Solve for δu_c,
  $u_c + u_f + \delta u_c$ is exact

Fine grid patch on flame improves the solution over the entire domain

Syncs for low Mach Algorithm

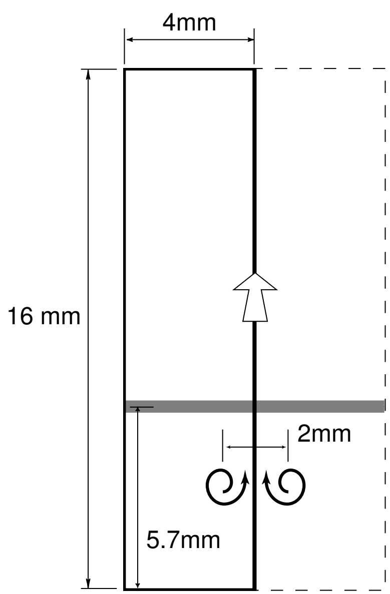
At syncronization point in subcyling procedure

1. **Re-advect:** Godunov velocities $U_C^{n+1/2} \neq$ Avg($U_F^{n+1/2}$).
   Find (global) δU_C, increment advection fluxes
2. **Reflux:** Set coarse grid fluxes $\Gamma_C = \sum \sum \Gamma_F$ on $c - f$.
   Interpolate coarse corrections to fine grid.
3. **Sync Project:** Corrected $\vec{U}$ must satify $\nabla \cdot \vec{U} = S$ over the composite grid.
4. **Average Down:** Conservative averaging, improves subsequent coarse grid advance

Resulting Algorithm is:

- Consistent data across levels
- Global conservation
- $\nabla \cdot U = S$ everywhere

Validation - AMR vs. Uniform



Problem setup:

- Flat 2D flame initialized with 1D solution
- Symmetry L-R, inflow bottom, outflow top
- Vortex pair superimposed on upward flow
- Vortex self-induced motion upward at 3.1 m/s
- Integrate until vortices tear through flame
- Compare uniform grid and adaptive solutions
  - Uniform $\Delta x = 31.25\ \mu$m
  - Adaptive $\Delta x_j =$ (125,62.5,31.25) μm

H_2O_2

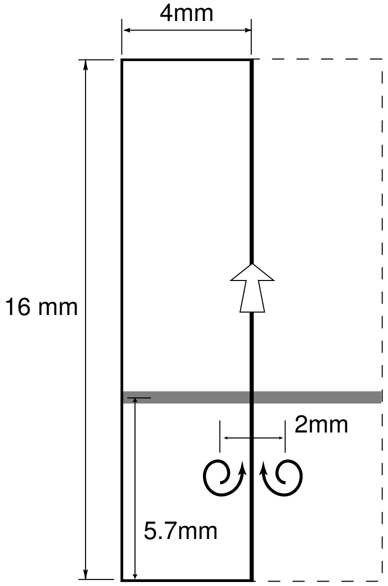
AMR solution

Results:

- Two factors-of-two refinement, AMR case $3\times$ faster
- Moving fine grids track vorticity, H_2O_2
- The adaptive and single-grid results are in excellent agreement
  - Flame position, detailed profile structure of flame intermediates
  - Velocity profiles (elliptic velocity matching)
  - No “imprints” in the solution where prior fine grid derefined

Day & Bell, CTM **4** (4) 2000

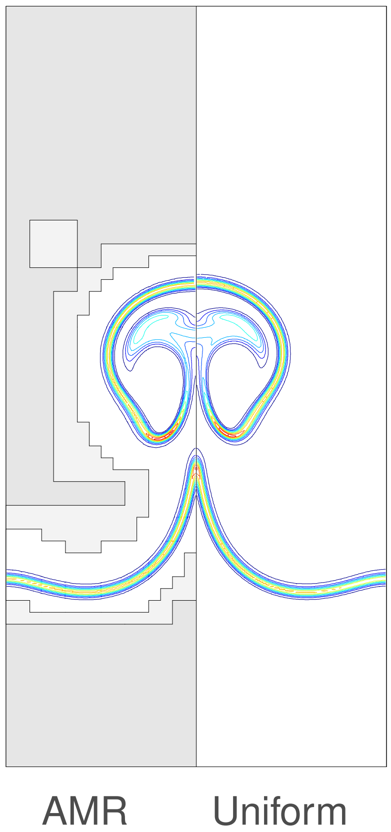
Validation - AMR vs. Uniform



Problem setup:

- Flat 2D flame initialized with 1D solution
- Symmetry L-R, inflow bottom, outflow top
- Vortex pair superimposed on upward flow
- Vortex self-induced motion upward at 3.1 m/s
- Integrate until vortices tear through flame
- Compare uniform grid and adaptive solutions
  - Uniform $\Delta x = 31.25\ \mu$m
  - Adaptive $\Delta x_j = (125{,}62.5{,}31.25)\ \mu$m

H_2O_2 at 85 μs



Results:

- Two factors-of-two refinement, AMR case 3$\times$ faster
- Moving fine grids track vorticity, H_2O_2
- The adaptive and single-grid results are in excellent agreement
  - Flame position, detailed profile structure of flame intermediates
  - Velocity profiles (elliptic velocity matching)
  - No “imprints” in the solution where prior fine grid derefined

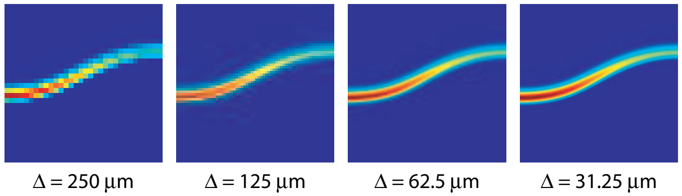
Day & Bell, CTM **4** (4) 2000

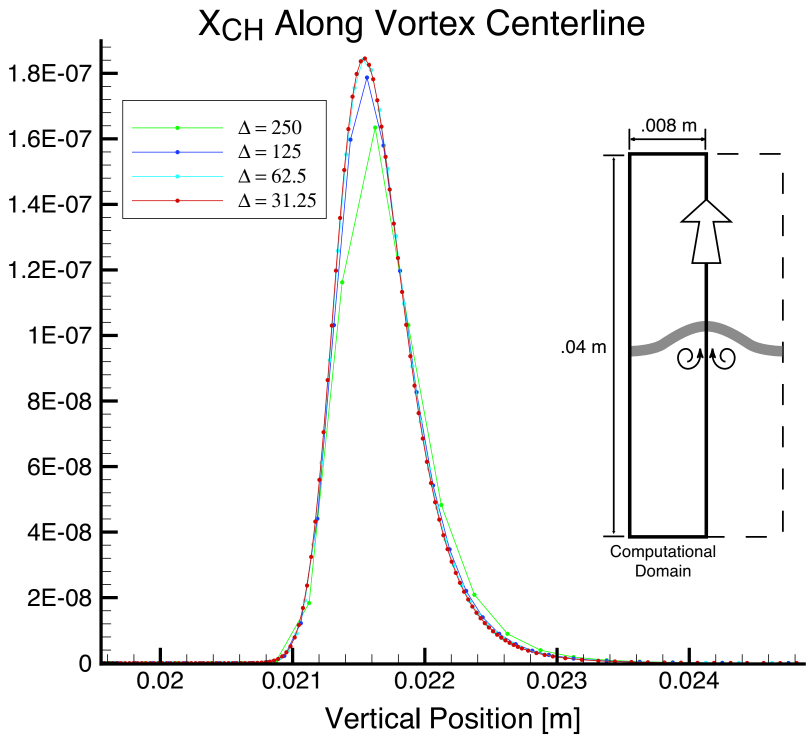
Resolution Requirements

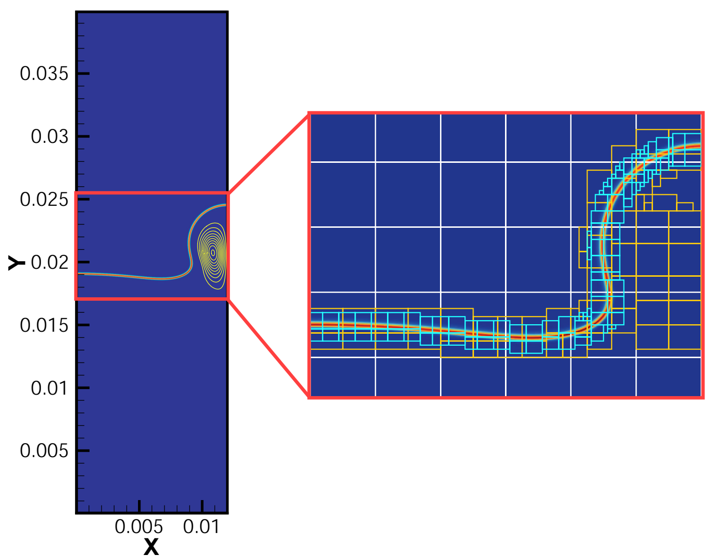
Use vortex-flame interaction experiments to determine flame resolution requirements

Similar to previous case

- Fuel is N_2-diluted CH_4/air
- Mechanism is GRI-Mech 1.2
  – 32 species, 177 reactions
- CH is trace species at flame







Representative adaptive solution

Dynamic Load-Balancing

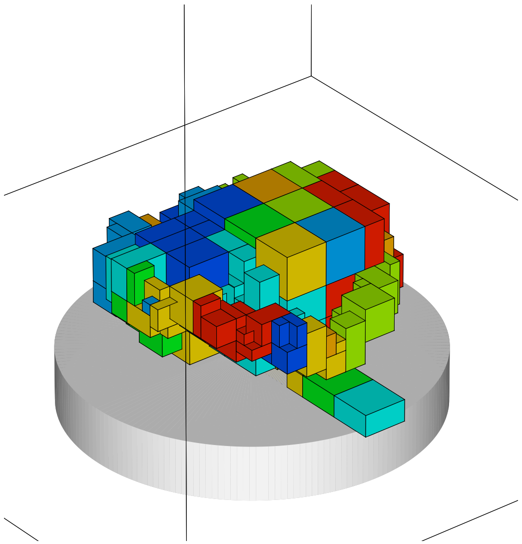
Approach: Estimate work per grid, distribute using heuristic KNAPSACK algorithm

Cells/grid often a good work estimate, but chemical kinetics may be highly variable

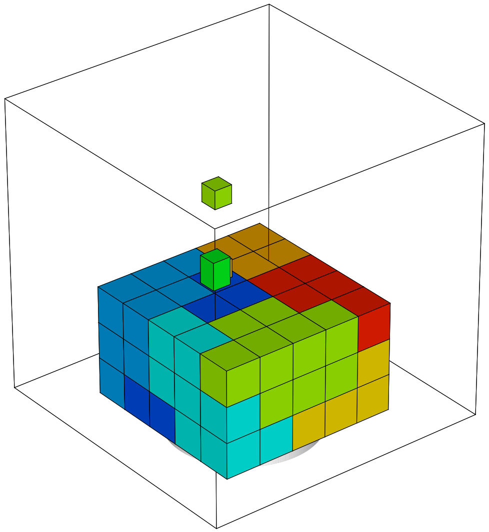
- Monitor chemistry integration work: count rate evaluations during the fluid time step
- Distribute chemistry work based on this work estimate (optional)

Parallel Communication: AMR data communication patterns are complex

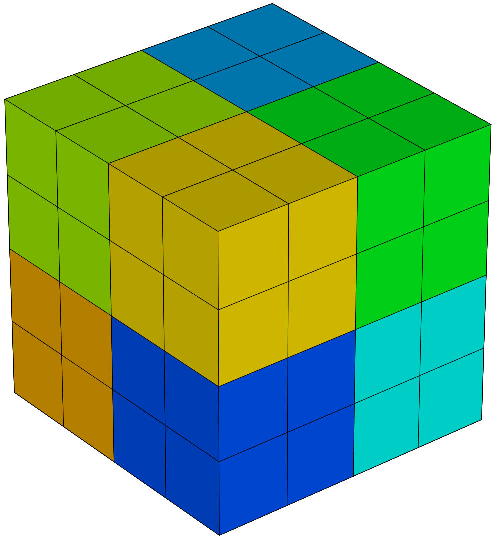
- Easy: distribute grids at a single level, minimize off-processor communication
- Hard: Incorporate coarse-fine interpolation (also, “recursive” interpolation)



Level 2



Level 1



Level 0

Results: Turbulence flame sheet

Three-dimensional isotropic turbulence propagating into a premixed flame

- Rutland and Zhang (1995) 1-step, DNS
- Tanahashi, et al (2000) Hydrogen, DNS
- Bell, et al (2002) Methane, low Mach

Flame:

- $\phi = 0.8$
- $\delta_L = 0.53$mm
- $S_L = 25$cm/s

Turbulence:

- $\ell_t = 1.0$mm
- $u'/S_L = 1.7, 4.3$

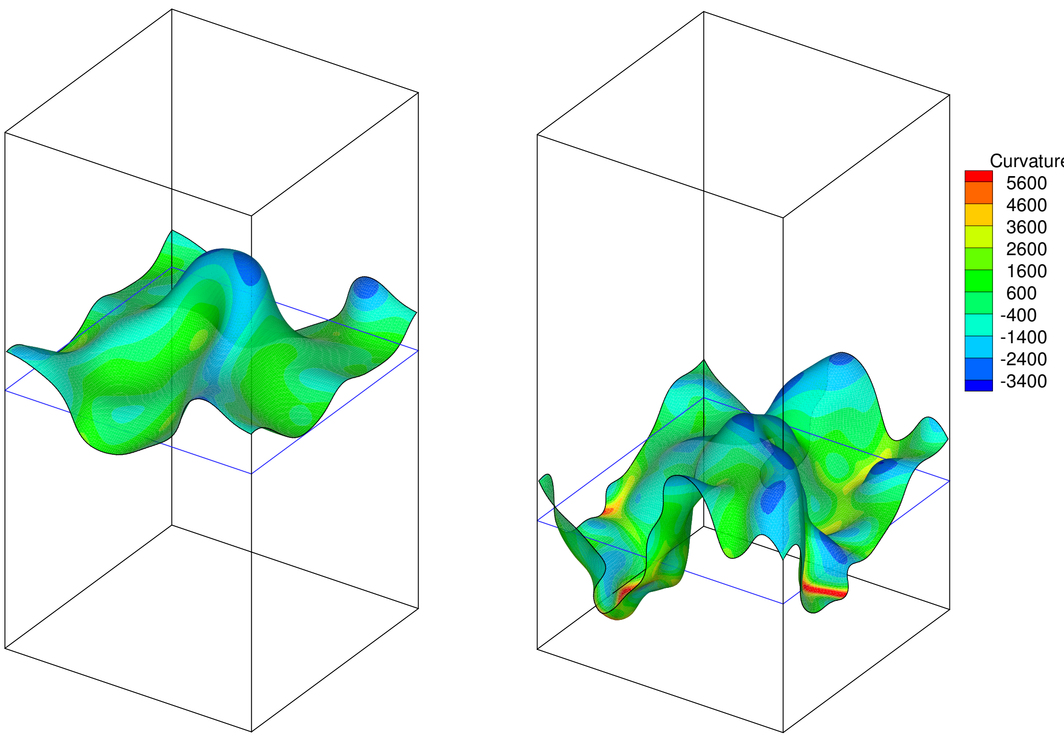
Computations:

- 8×8×16 mm domain
- doubly periodic
- $\Delta x_{\text{eff}} = 62.5\mu$m

Model:

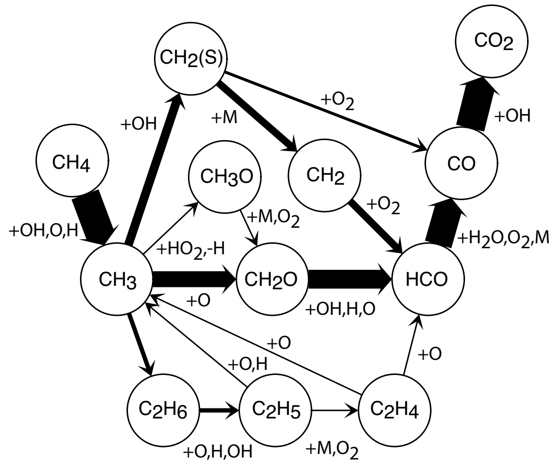
- DRM-19
- 20 species/84 reacs

$T = 1500K$ surfaces, colored by mean curvature.

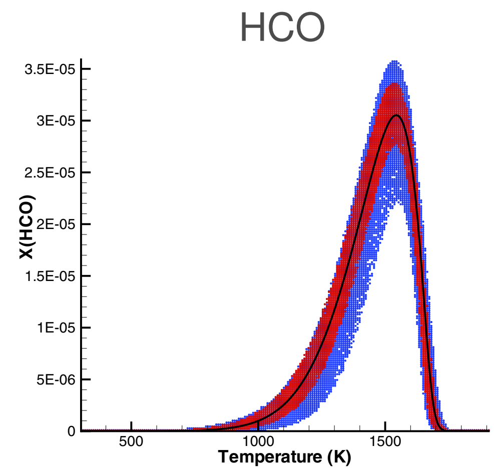


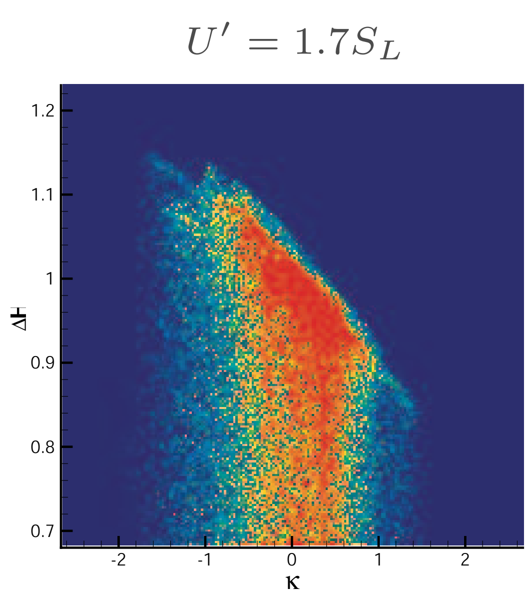
$u'/S_L = 1.7$ $\qquad$ $u'/S_L = 4.3$

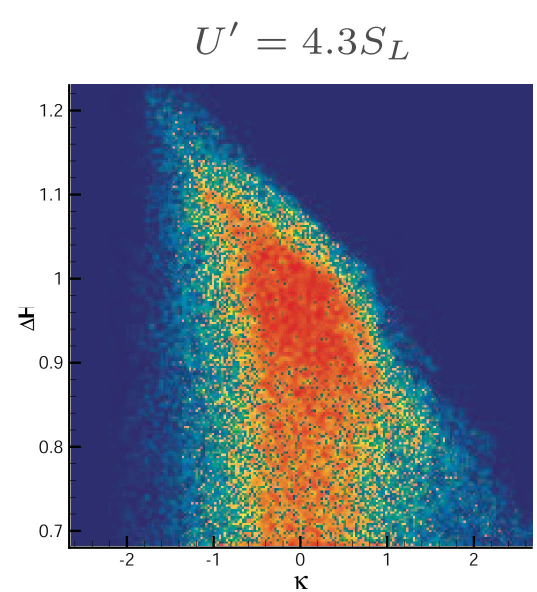
Heat Release and Flame Curvature



Carbon Flow Network





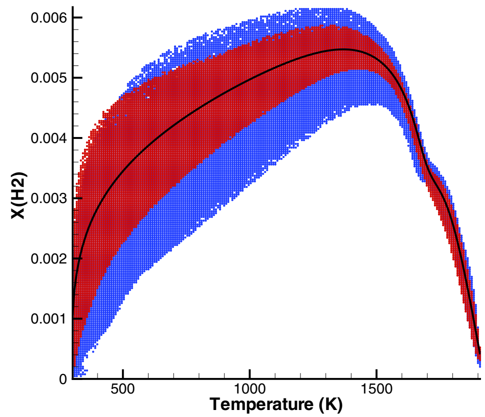


Heat release vs. Curvature

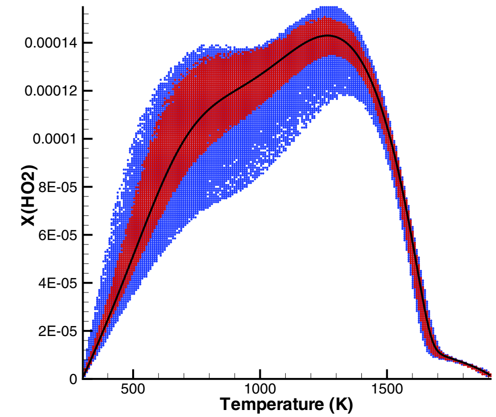
(ΔH, κ normalized by ΔH_L, δ_L)

Redistribution of Species

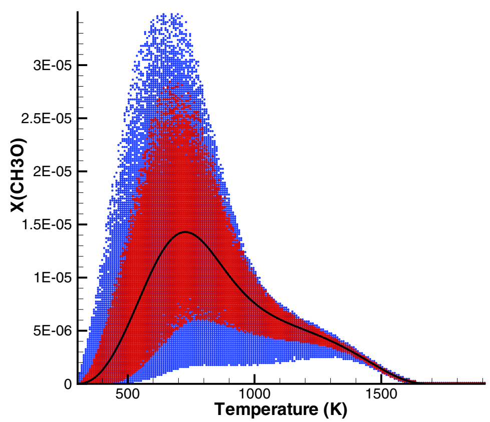
H_2



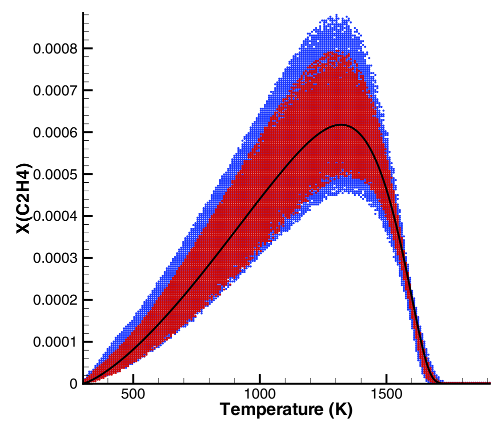
HO_2



CH_3O

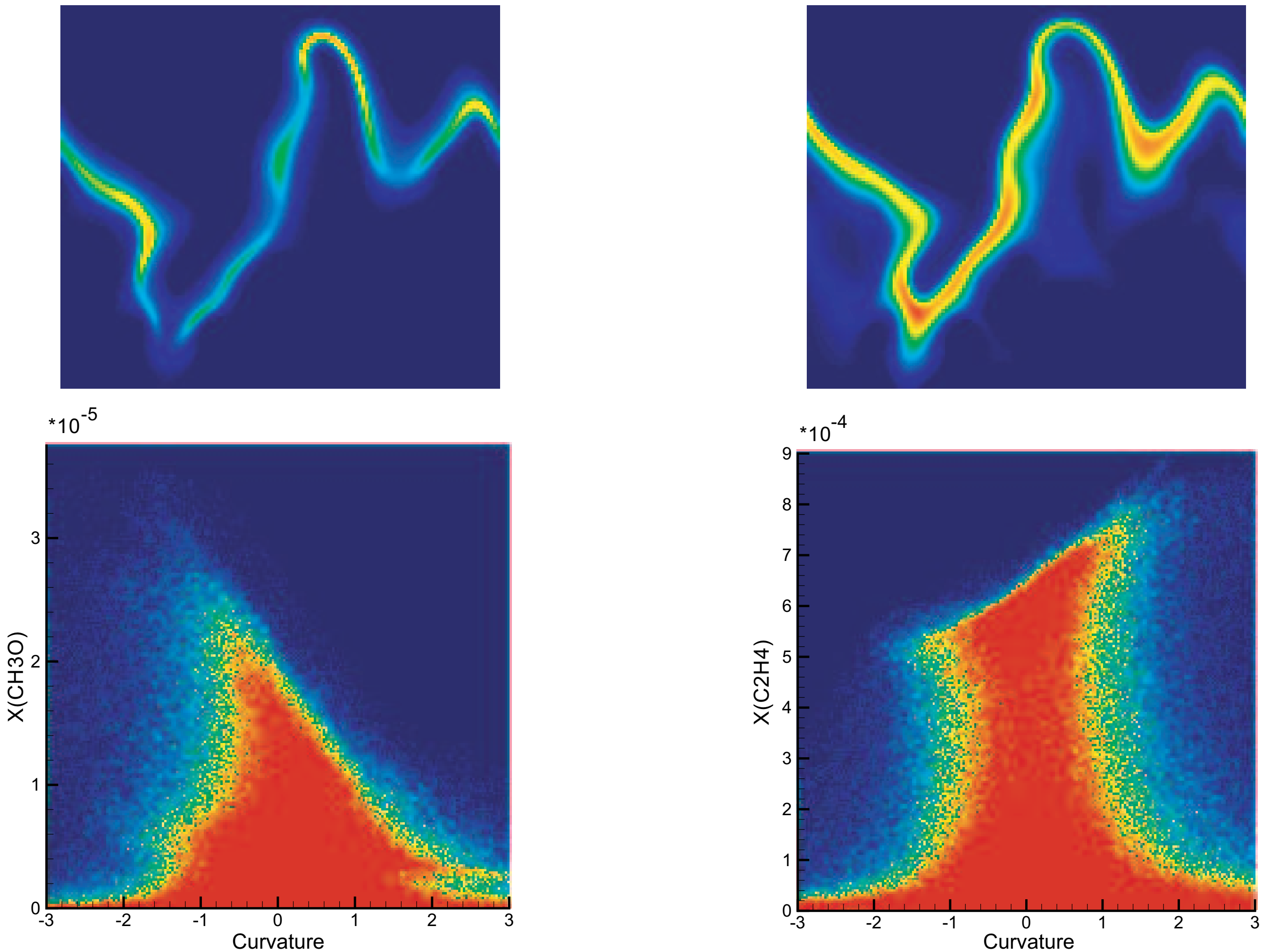


C_2H_4



Turbulence chemistry interaction

CH_3O C_2H_4



Species concentration versus flame curvature

Laboratory Flames

Rod-stabilized Flame

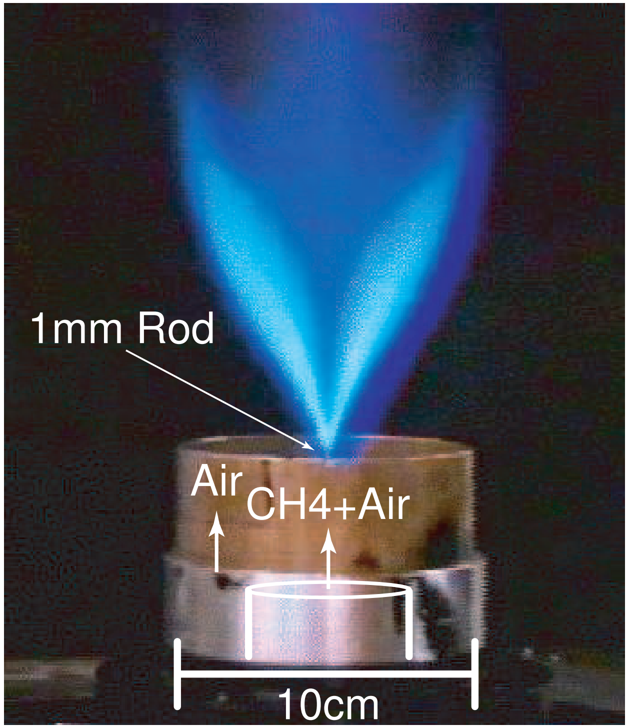


Photo courtesy R. Cheng

Can we make the link between flames that can be simulated in 3D and those that can be observed in the lab?

WORK-IN-PROGRESS

Rod-stabilized V-flame:

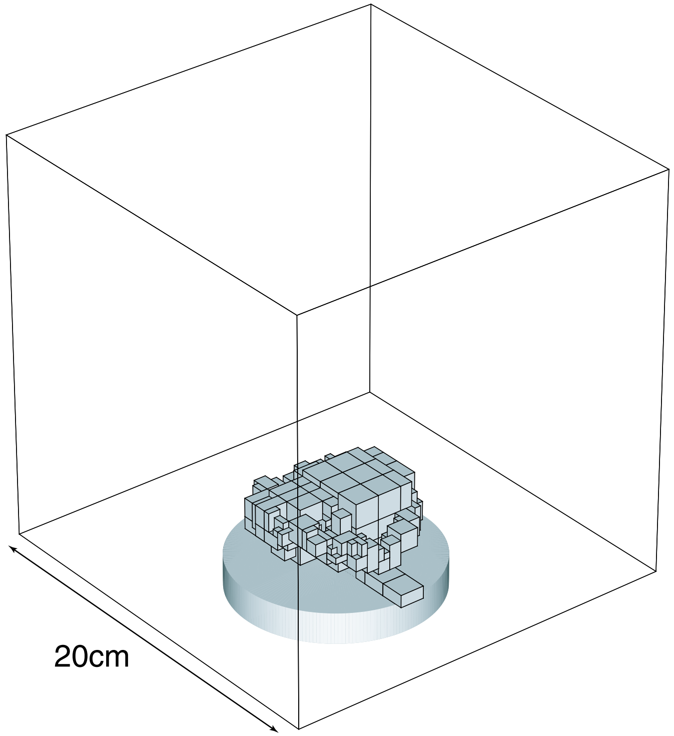
Turbulent ($\ell_t = 5$mm, $I = 5$%) premixed fuel ($\phi = 0.75$) flows past 1 mm rod at about 1 m/s. Concentric fuel/air coflow with “tophat” profiles in the mean and fluctuating velocity components. Requires domain of 10-20 cm to avoid boundary effects.

This is a rather daunting task, can we pull it off?

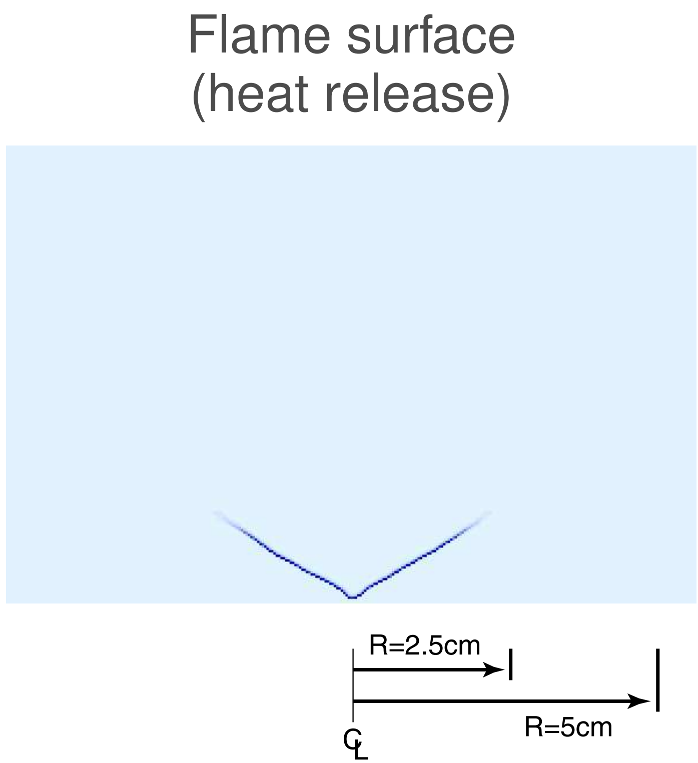
Begin with a simple 2-step methane mechanism, and previous experience, in terms of resolution requirements to correctly compute the flame position and large eddies that affect flame propagation.

After establishing statistically stationary flame, add refinement and chemical/transport detail (e.g. DRM-19)

Early Results



Finest grids in a preliminary
3-level simulation



Vertical plane through center
perpendicular to rod

NOTE: This flame is “wake stabilized”. The simulation introduced a small pocket of cool air on top of the rod that cools the flame. First rod-stabilized results still in the queue as I speak (!)

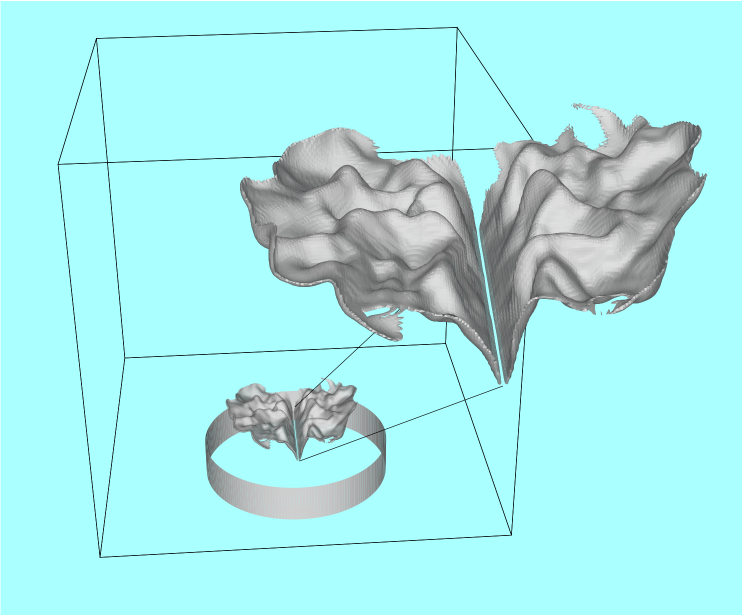
Summary and Future Work

Summary

Presented an algorithm for low Mach number combustion that is

- Adaptive
- Conservative
- Second-order in time and space

The algorithm has been validated extensively and is suitable for aggressive application on large-scale reacting flow problems using existing computational hardware.



Isopleth of CO mass fraction using DRM-19

Future Work

Use the problems discussed to drive further development of algorithms and analysis tools for studying complex laboratory-scale reacting flows.